

Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Original) A method of determining an edge for inversion in a cyclic compound directed graph, comprising:
 - evaluating a plurality of graph nodes of a graph for generation of a first node order subset pair;
 - determining the graph requires recursive evaluation for completing node ordering of the plurality of graph nodes;
 - dividing the graph into a plurality of graph partitions; and
 - generating respective node order subset pairs of the graph partitions.
2. (Original) The method of claim 1, wherein the step of evaluating further includes:
 - identifying a removable sink node of the graph;
 - removing the removable sink node from the graph; and
 - appending a designation associated with the removable sink node in a first node order subset of the first node order subset pair.
3. (Original) The method of claim 2, wherein the steps of identifying and removing are repeated until no removable sink nodes remain in the graph.
4. (Original) The method of claim 2, wherein the step of evaluating further includes:
 - identifying a removable source node of the graph;
 - removing the removable source node from the graph; and
 - appending a designation associated with the removable source node in a second node order subset of the first node order subset pair.
5. (Original) The method of claim 4, wherein the steps of identifying and removing the removable source node are repeated until no removable source nodes remain in the graph.
6. (Currently Amended) The method of claim 4, wherein: ~~the step of determining further includes:~~
the subsets of each subset pair are respectively concatenated, results of the concatenation of subset pairs respectively associated with the partitions are appended to the second node order subset; and

the first subset and the second subset are concatenated, after said results are appended to provide a node order set for use in determining an edge for inversion.

identifying a plurality of top-level subgraph nodes in graph elements remaining after removal of the removable sink node and the removable source node.

7. (Currently Amended) A computer program product in a computer readable medium for identifying a graph edge for inversion in a cyclic compound directed graph, the computer program product comprising:

first instructions for receiving an input graph[[:]], wherein the input graph is a data structure defining a plurality of graph edges, each comprising a respective source node and a target node;

second instructions for initializing a first subset pair and iteratively identifying and removing removable sink nodes and removable source nodes from the input graph, the removable sink nodes recorded in a first subset of the first subset pair and the removable source nodes recorded in a second subset of the first subset pair;

third instructions, responsive to removal of the removable sink nodes and removable source nodes by the second instructions, that identify remaining graph elements; and

fourth instructions, responsive to the third instructions identifying the remaining graph elements, adapted to divide the remaining graph elements into graph partitions, wherein the second instructions initialize and record removable sink nodes and removable source nodes in respective subset pairs for each of the graph partitions[[:]];

fifth instructions for generating a node order set including each node of the graph, wherein the node order set is generated from the first subset pair and the subset pairs associated with the partitions;

sixth instructions for associating a respective sequence number with each of the nodes of the node order set; and

seventh instructions for comparing a sequence number of a source node of a first edge defined in a record of the data structure with a sequence number of a target node of the first edge, the seventh instructions, responsive to determining that the sequence number of the source node of the first edge is larger than the sequence number of the target node of the first edge, identify the first edge as an inversion edge.

8. (Original) The computer program product of claim 7, wherein the remaining graph elements include at least two top-level subgraph nodes.

9. (Original) The computer program product of claim 7, further including:
fifth instructions that concatenate respective subsets of each subset pair.
10. (Original) The computer program product of claim 9, wherein results of the concatenation of
subset pairs associated with the partitions are appended to the second subset.
11. (Original) The computer program product of claim 10, wherein the first subset and the second
subset are concatenated after appendage of the concatenation results.
12. Canceled
13. Canceled
14. Canceled
15. (Currently Amended) The computer program product of claim ~~14~~ 7, wherein the seventh
instructions, responsive to identifying the first edge as an inversion edge, insert an asserted inversion
value into the record.
16. (Currently Amended) The computer program product of claim ~~13~~ 7, wherein: ~~the input graph is a
data structure defining a plurality of graph edges each comprising a respective source node and a target
node, the computer program product further including~~
~~said seventh instructions, for comparing a sequence number of a source node of a first edge
defined in a record of the data structure with a sequence number of a target node of the first edge, the
seventh instructions, responsive to determining that the sequence number of the source node of the first
edge is less than the sequence number of the target node of the first edge, identify the first edge as a non-
inverted edge.~~
17. (Original) The computer program product of claim 16, wherein the seventh instructions,
responsive to identifying the first edge as a non-inverted edge, insert a non-asserted inversion value into
the record.
18. Canceled

19. Canceled

20. Canceled

21. (New) A method of determining an edge for inversion in a cyclic compound directed graph, said method comprising the steps of:

receiving an input graph comprising a data structure defining a plurality of graph edges, each comprising a respective source node and a target node;

evaluating a plurality of graph nodes of the input graph in order to generate a first node order subset pair;

dividing the input graph into a plurality of partitions;

generating a node order subset pair corresponding to each of the graph partitions;

generating a node order set including each node of the input graph, wherein the node order set is generated from the first subset pair and from the subset pairs corresponding to the partitions, a sequence number being associated with each of the nodes of the node order set; and

comparing the sequence number of a source node of a first edge defined in a record of the data structure with a sequence number of a target node of the first edge, in order to identify the first edge as an inversion edge when the sequence number of the first edge source node is larger than the sequence number of the first edge target node.

22. (New) The method of Claim 21, wherein:

in response to identifying the first edge as an inversion edge, an asserted inversion value is inserted into the record.

23. (New) The method of Claim 21, wherein the input graph is a data structure defining a plurality of graph edges, each comprising a respective source node and a target node, and the method further includes the step of:

comparing a sequence number of a source node of a first edge defined in a record of the data structure with a sequence number of a target node of the first edge, and responsive to determining that the sequence number of the source node of the first edge is less than the sequence number of the target node of the first edge, identifying the first edge as a non-inverted edge.

24. (New) The method of Claim 23, wherein:

in response to identifying the first edge as a non-inverted edge, a non-asserted inversion value is inserted into the record.